

This listing of claims will replace all prior versions, and listings, of claims in the application:

**Listing of Claims**

1. (Currently Amended) A method for input parameter binding, comprising:

at bind time, storing optimization information in a bind-in structure wherein the bind-in structure has an associated section number;

when executing a statement, when performing bind-in of host variables, comparing data in an application structure received with the statement with optimization information in ~~[[a]] the~~ bind-in structure, ~~wherein the application structure includes data to be inserted into a data store and~~ wherein the optimization information includes at least one of data type, length, Coded Character Set Identifier, an array size, an indication of whether conversions are required, and an indication of whether the required conversions are valid; ~~[[and]]~~

when there is a match between the data in the application structure and data in the optimization information in the bind-in structure, executing the statement with the optimization information to perform one of fetching data from the data store and inserting data into the data store, wherein the bind-in structure and the statement have a same section number; and

when there is not a match between the data in the application structure and the optimization information,

regenerating optimization information; and

executing the statement with the regenerated optimization information to perform one of fetching data from the data store and inserting data into the data store.

2. (Cancelled)

3. (Original) The method of claim 1, further comprising:

at bind time, storing the optimization information in the bind-in structure.

4. (Cancelled)

5. (Original) The method of claim 1, further comprising:

for fixed length data,  
storing an increment length by which a data pointer that is pointing to data in an application program area is to be incremented to find a location of a next data value; and  
calculating the location of the next data value by adding the increment length to the data pointer.

6. (Original) The method of claim 1, further comprising:  
for distributed processing, at a client computer, calculating a location of data in a client communications buffer.

7. (Original) The method of claim 1, further comprising:  
for distributed processing, at a server computer, calculating a location of data in a server communications buffer.

8. (Original) The method of claim 1, further comprising:  
for distributed processing, at a client computer, calculating a location of data in an application program address space.

9. (Original) The method of claim 1, further comprising:  
when returning a handle to a cursor to a result set from a stored procedure to an application, recalculating the optimization information.

10. (Currently Amended) A method for output parameter binding, comprising:  
at bind time, storing optimization information in a bind-out structure wherein the bind-out structure has an associated section number;  
when executing a statement, when performing bind-out of host variables, comparing data in an application structure received with the statement with optimization information in [[a]] the bind-out structure, wherein the application structure is capable of storing data to be retrieved from a data store and wherein the optimization information includes at least one of data type, length, Coded Character Set Identifier, an array size, an indication of whether conversions are required, and an indication of whether the required conversions are valid; [[and]]

when there is a match between the data in the application structure and data in the optimization information in the bind-out structure, executing the statement with the optimization information to perform one of fetching data from the data store and inserting data into the data store, wherein the bind-out structure and the statement have a same section number; and

when there is not a match between the data in the application structure and the optimization information,  
regenerating optimization information; and  
executing the statement with the regenerated optimization information to perform one of fetching data from the data store and inserting data into the data store.

11. (Cancelled)

12. (Original) The method of claim 10, further comprising:  
at bind time, storing the optimization information in the bind-out structure.

13. (Cancelled)

14. (Original) The method of claim 10, further comprising:  
for fixed length data,  
storing an increment length by which a data pointer that is pointing to data in an application program area is to be incremented to find a location of a next data value; and  
calculating the location of the next data value by adding the increment length to the data pointer.

15. (Original) The method of claim 10, further comprising:  
for distributed processing, at a client computer, calculating a location of data in a client communications buffer.

16. (Original) The method of claim 10, further comprising:  
for distributed processing, at a server computer, calculating a location of data in a server communications buffer.

17. (Original) The method of claim 10, further comprising:  
for distributed processing, at a client computer, calculating a location of data in an application program address space.

18. (Original) The method of claim 10, further comprising:  
when returning a handle to a cursor to a result set from a stored procedure to an application, recalculating the optimization information.

19. (Currently Amended) An article of manufacture including a program for input parameter binding, wherein the program causes operations to be performed, the operations comprising:

at bind time, storing optimization information in a bind-in structure wherein the bind-in structure has an associated section number;

when executing a statement, when performing bind-in of host variables, comparing data in an application structure received with the statement with optimization information in ~~[[a]] the bind-in structure, wherein the application structure includes data to be inserted into a data store~~ and wherein the optimization information includes at least one of data type, length, Coded Character Set Identifier, an array size, an indication of whether conversions are required, and an indication of whether the required conversions are valid; ~~[[and ]]~~

when there is a match between the data in the application structure and data in the optimization information in the bind-in structure, executing the statement with the optimization information to perform one of fetching data from the data store and inserting data into the data store, wherein the bind-in structure and the statement have a same section number; and

when there is not a match between the data in the application structure and the optimization information,

regenerating optimization information; and  
executing the statement with the regenerated optimization information to perform one of fetching data from the data store and inserting data into the data store.

20. (Cancelled)

21. (Original) The article of manufacture of claim 19, wherein the operations further comprise:

at bind time, storing the optimization information in the bind-in structure.

22. (Cancelled)

23. (Original) The article of manufacture of claim 19, wherein the operations further comprise:

for fixed length data,

storing an increment length by which a data pointer that is pointing to data in an application program area is to be incremented to find a next data value; and

calculating the location of the next data value by adding the increment length to the data pointer.

24. (Original) The article of manufacture of claim 19, wherein the operations further comprise:

for distributed processing, at a client computer, calculating a location of data in a client communications buffer.

25. (Original) The article of manufacture of claim 19, wherein the operations further comprise:

for distributed processing, at a server computer, calculating a location of data in a server communications buffer.

26. (Original) The article of manufacture of claim 19, wherein the operations further comprise:

for distributed processing, at a client computer, calculating a location of data in an application program address space.

27. (Original) The article of manufacture of claim 19, wherein the operations further comprise:

when returning a handle to a cursor to a result set from a stored procedure to an application, recalculating the optimization information.

28. (Currently Amended) An article of manufacture including a program for output parameter binding, wherein the program causes operations to be performed, the operations comprising:

at bind time, storing optimization information in a bind-out structure wherein the bind-out structure has an associated section number;

when executing a statement, when performing bind-out of host variables, comparing data in an application structure received with the statement with optimization information in [[a ]] the bind-out structure, ~~wherein the application structure is capable of storing data to be retrieved from a data store and~~ wherein the optimization information includes at least one of data type, length, Coded Character Set Identifier, an array size, an indication of whether conversions are required, and an indication of whether the required conversions are valid;[[ and]]

when there is a match between the data in the application structure and data in the optimization information in the bind-out structure, executing the statement with the optimization information to perform one of fetching data from the data store and inserting data into the data store, wherein the bind-out structure and the statement have a same section number; and

when there is not a match between the data in the application structure and the optimization information,

regenerating optimization information; and  
executing the statement with the regenerated optimization information to perform one of fetching data from the data store and inserting data into the data store.

29. (Cancelled)

30. (Original) The article of manufacture of claim 28, wherein the operations further comprise:

at bind time, storing the optimization information in the bind-out structure.

31. (Cancelled)

32. (Original) The article of manufacture of claim 28, wherein the operations further comprise:

for fixed length data,

storing an increment length by which a data pointer that is pointing to data in an application program area is to be incremented to find a next data value; and

calculating the location of the next data value by adding the increment length to the data pointer.

33. (Original) The article of manufacture of claim 28, wherein the operations further comprise:

for distributed processing, at a client computer, calculating a location of data in a client communications buffer.

34. (Original) The article of manufacture of claim 28, wherein the operations further comprise:

for distributed processing, at a server computer, calculating a location of data in a server communications buffer.

35. (Original) The article of manufacture of claim 28, wherein the operations further comprise:

for distributed processing, at a client computer, calculating a location of data in an application program address space.

36. (Original) The article of manufacture of claim 28, wherein the operations further comprise:

when returning a handle to a cursor to a result set from a stored procedure to an application, recalculating the optimization information.

37. (Currently Amended) A system for input parameter binding, comprising:  
at bind time, storing optimization information in a bind-in structure wherein the bind-in structure has an associated section number;

when executing a statement, when performing bind-in of host variables, means for comparing data in an application structure received with the statement with optimization information in ~~[[a]] the bind-in structure, wherein the application structure includes data to be inserted into a data store and~~ wherein the optimization information includes at least one of data type, length, Coded Character Set Identifier, an array size, an indication of whether conversions are required, and an indication of whether the required conversions are valid; ~~[[and]]~~

when there is a match between the data in the application structure and data in the optimization information in the bind-in structure, means for executing the statement with the optimization information to perform one of fetching data from the data store and inserting data into the data store, wherein the bind-in structure and the statement have a same section number;  
and

when there is not a match between the data in the application structure and the optimization information,  
regenerating optimization information; and  
executing the statement with the regenerated optimization information to perform one of fetching data from the data store and inserting data into the data store.

38. (Currently Amended) A system for output parameter binding, comprising:  
at bind time, storing optimization information in a bind-out structure wherein the bind-out structure has an associated section number;

when executing a statement, when performing bind-out of host variables, means for comparing data in an application structure received with the statement with optimization information in ~~[[a ]] the bind-out structure, wherein the application structure is capable of storing data to be retrieved from a data store and~~ wherein the optimization information includes at least one of data type, length, Coded Character Set Identifier, an array size, an indication of whether conversions are required, and an indication of whether the required conversions are valid; ~~[[and]]~~

when there is a match between the data in the application structure and data in the optimization information in the bind-out structure, means for executing the statement with the



optimization information to perform one of fetching data from the data store and inserting data into the data store, wherein the bind-out structure and the statement have a same section number;  
and

when there is not a match between the data in the application structure and the optimization information,

regenerating optimization information; and  
executing the statement with the regenerated optimization information to perform one of fetching data from the data store and inserting data into the data store.